VILNIUS UNIVERSITY

FACULTY OF MATHEMATICS AND INFORMATICS

ARTIFICIAL NEURAL NETWORKS COURSE

SATELLITE IMAGE RECOGNITION TASK

2st Report

Author: Edvardas Timoscenka 3rd year student of Mathematics and Mathematical Applications

Vilnius

2023

Contents

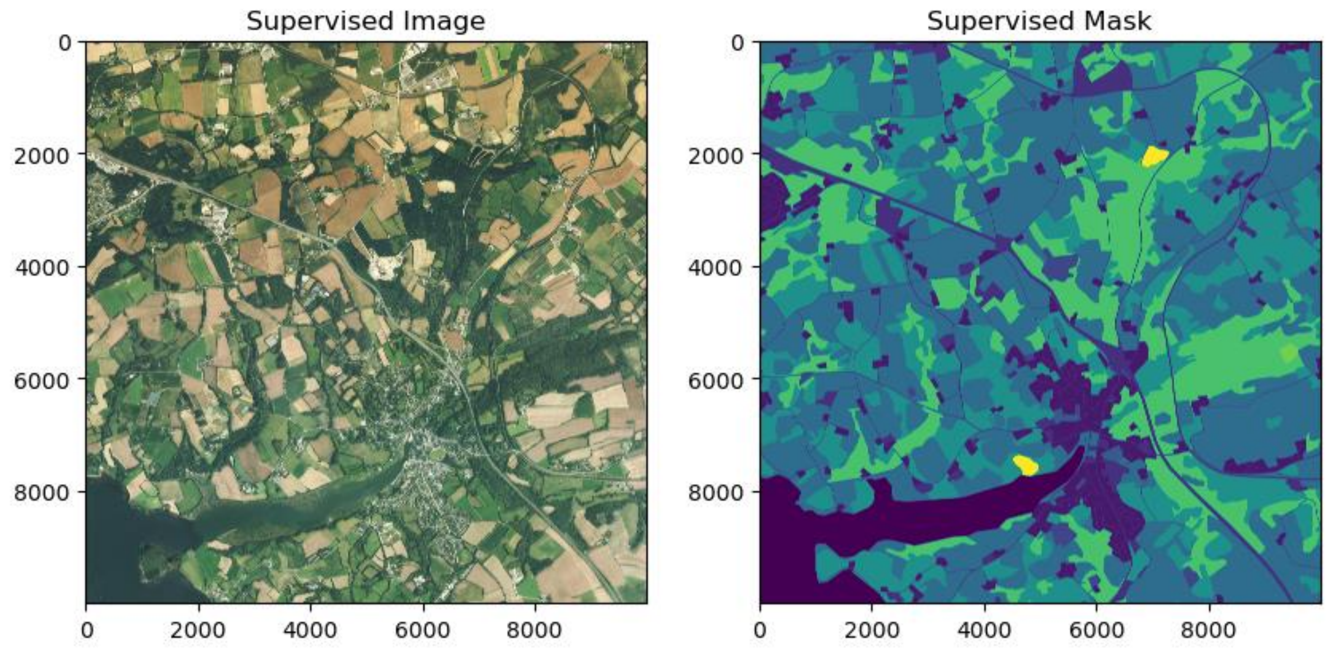
1. Images and data.

1.1. Original Dataset Images and Masks.

The original dataset comprised high-resolution images, each measuring 10,000 pixels by 10,000 pixels, with RGB color values. Notably, the dataset was predominantly centered around the Brest region, featuring a total of 377 images. In contrast, the Le Mans region had the lowest representation in the dataset with only 226 images.

Furthermore, the dataset included corresponding image masks, maintaining the same dimensions, but with pixel values indicating class labels. Figure 1 presents an illustrative image-mask pair.

Figure 1. Image and its mask pair in the original dataset.

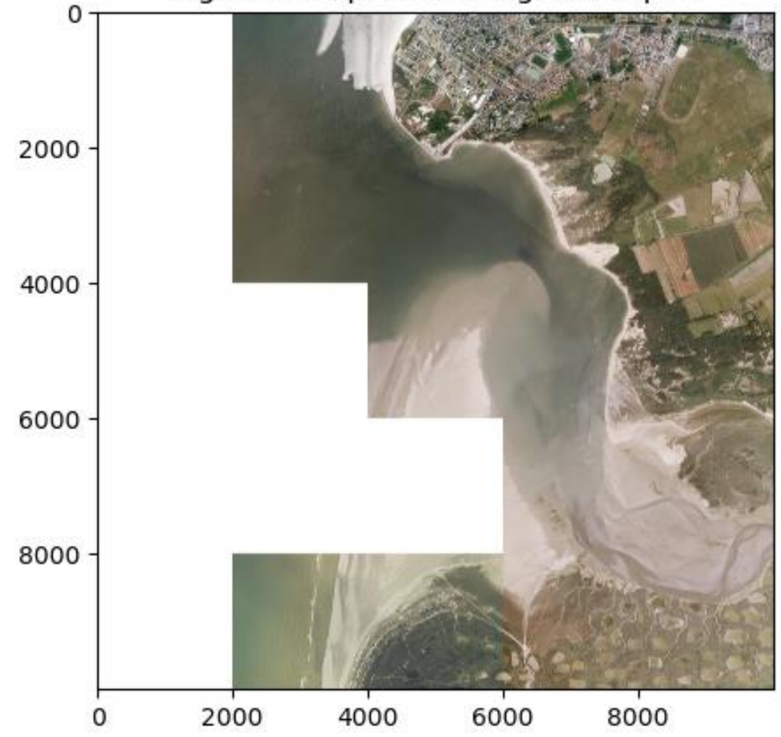


The miniFrance dataset was designed for semi-supervised learning, which implies that not every image had an associated mask. Approximately 53% of the images had corresponding masks, leaving the remainder unlabeled.

The St. Brieuc region had the lowest mask coverage, with only 25% of its images labeled, while the Nice region had the highest proportion of labeled masks, encompassing approximately 77% of its images.

In summary, the dataset encompassed a total of 2,816 images and 1,490 corresponding masks. It's worth noting that some images exhibited imperfections in terms of quality, featuring missing data in certain areas, as visualized in Figure 2.

Figure 2. Imperfect image example.



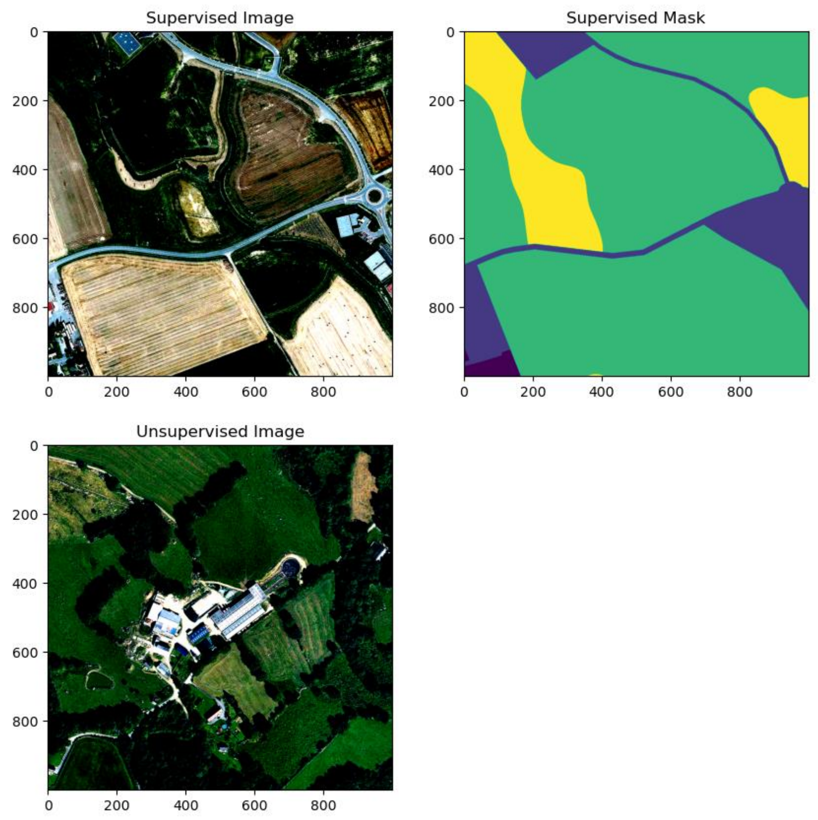
1.2. Data preprocessing.

As outlined in the first report, the original dataset underwent modifications to enhance its computational feasibility and facilitate data manipulation. Each original image and its corresponding mask were partitioned into smaller sub-images, each measuring 1,000 pixels by 1,000 pixels, preserving the original values. Subsequently, a random selection process was employed to choose sub-images from each source image, which were then stored in the designated data repository for future use. To address variations in image quality, an additional filtering step was introduced to assess the percentage of white pixels within each sub-image.

The filter examined whether a sub-image was present in over 90% of the source images, and if not, the random selection process was repeated using the same source image. This procedure aimed to minimize the inclusion of low-quality images in the dataset for subsequent model training.

A visual representation of the results can be found in Figure 3.

Figure 3. Images and mask in the final dataset.



A subsequent round of preprocessing was executed following the same methodology, expanding the image count to exceed 3,000 images. After the preprocessing phase, the dataset encompassed a total of 3,477 images and 1,784 corresponding masks.

The new dataset consisting of sub-images and sub-masks had the following class distribution (calculated from supervised training dataset):

Table 1: Class distribution in supervised dataset versus goal distribution.

Class	% pixels	Goal	p.p. Difference
No information	31.8%	-	-
Urban fabric	6.2 %	9.9 %	3.7 %
Industrial, commercial, public, military, private and transport units	4.1 %	6.5 %	2.4 %
Mine, dump, and construction sites	0.4 %	0.7 %	0.3 %
Artificial non-agricultural vegetated areas	0.9 %	1.2 %	0.3 %
Arable land (annual crops)	18.7 %	30.7 %	12.0 %
Permanent crops	0.4 %	1.3 %	0.9 %
Pastures	21.0 %	27.3 %	6.3 %
Complex and mixed cultivation patterns	0.0 %	0.0 %	0.0 %
Orchards at the fringe of urban classes	0.0 %	0.0 %	0.0 %
Forests	12.5 %	16.0 %	3.5 %
Herbaceous vegetation associations	1.9 %	4.5 %	2.6 %
Open spaces with little or no vegetation	0.9 %	0.1 %	0.8 %
Wetlands	0.6 %	0.7 %	0.1 %
Water	0.6 %	1.0 %	0.4 %
Clouds and shadows	0.1 %	0.1 %	0.0 %

1.3. Data preparation for the model.

A dedicated Data Loader was created to manage both the supervised and unsupervised data streams intended for the model. This loader creates random samples from the dataset, ensuring a specified proportion of supervised and unsupervised data to be fed into the model. Moreover, the Data Loader performs transformations on the images, incorporating mean and standard deviation adjustments required for compatibility with the DeepLabV3 models and recognizing the model's expectation of input data in the form of minibatches and different shape, the Data Loader is designed to handle this process.

2. Neural Networks architecture.

As of right now, no planned architecture has been fully implemented, with the exception of an experimental DeepLabV3 model employing a ResNet-50 backbone. This model was deployed primarily for the purpose of testing and evaluating the data loader's performance. Additionally, it served as a preliminary assessment of the model's capabilities on a personal computer setup.

3. Computational resource usage.

As planned before, my personal computer is used for carrying out most of the research and implementation. Through the testing model, a significant bottleneck has been identified in this approach - the GPU's VRAM. Presently, the GPU in use has 24GB of VRAM, which still faces challenges when handling the DeepLabV3 model, given its substantial size. The biggest issue arises when attempting to employ larger batch sizes as input into the model. For instance, with the batch parameter set to 5, the model demands approximately 35GB of VRAM, exceeding the available capacity. The most optimal batch size found now is 2, but there is potential for increasing this batch size through the implementation of efficient memory management.

4. Plans for other reports.

The next phase of the project involves the complete deployment of the DeepLabV3 model, integrating it into the existing system, and commencing work on the development of the U-NET model.

5. Project code.

The project code can be found in the GitHub repository located here:
https://github.com/Efoks/sem_sat_img_segmentation